REPORT DOCUMENTATION PAGE

Form Approved
OMB NO. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE 9 October 2000	3. REPORT TYPE AND DATES COVERED Final Report

4. TITLE AND SUBTITLE Analysis and Simulation of Radiative Transfer in the Presence of Non-Lambertian Surfaces	5. FUNDING NUMBERS DAAH04-96-1-0077
6. AUTHOR(S) James Arvo	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) California Institute of Technology Pasadena, CA 91125	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) U. S. Army Research Office P.O. Box 12211 Research Triangle Park, NC 27709-2211	10. SPONSORING / MONITORING AGENCY REPORT NUMBER ARO 35654.4-MA-YIP

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12 a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution unlimited.	12 b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This final report summarizes the work performed during the term of the grant. The research results are organized into four categories:

1. Distributed Monte Carlo Image Synthesis
2. Perturbation Methods for Specular Reflection
3. Interactive Symbolic Manipulation
4. Recovering Geometry from Range Data

14. SUBJECT TERMS 20001122 071	15. NUMBER OF PAGES
	16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION ON THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500

Standard Form 298 (Rev.2-89)
Prescribed by ANSI Std. 239-18
298-102

Analysis and Simulation of Radiative Transfer in the Presence of non-Lambertian Surfaces

Final Report

James Arvo

October 9, 2000

U.S. ARMY RESEARCH OFFICE

California Institute of Technology
Computer Science 256-80
Pasadena, CA 91125

APPROVED FOR PUBLIC RELEASE;
DISTRIBUTION UNLIMITED.

Contents

List of Figures

1 Problems Studied

This final report summarizes the work performed during the term of the grant. The research results are organized into four categories:

1. Distributed Monte Carlo Image Synthesis
2. Perturbation Methods for Specular Reflection
3. Interactive Symbolic Manipulation
4. Recovering Geometry from Range Data

The first three categories describe work that relates directly to the initial objectives of the project, which targeted the study of non-Lambertian reflection in image synthesis and the development of interactive techniques for natural manipulation of symbolic information. The fourth category describes work performed that was beyond the initial scope of the proposal; the goal of this project was to develop a robust method for geometry acquisition based on simple and readily available techniques for acquiring approximate range data.

The work performed on distributed Monte Carlo image synthesis focused primarily on load balancing schemes appropriate for rendering complex models on massively parallel systems. In particular, we have demonstrated the use of diffusion to propagate work to idle processors using a scheme that is applicable to arbitrary processor topologies. The method has been demonstrated on scenes consisting of tens of thousands of polygons exhibiting diffuse, glossy, and mirror reflection.

To gain a better understanding of specular (mirror-like) reflection in computer generated images, we investigated the use of perturbation methods in the context of ray tracing. We have shown how to compute the Taylor expansion of a reflection path, and how to apply such an expansion to the rapid computation of perturbed paths. We have demonstrated how perturbation methods can be used to render specular reflections in arbitrary curved surfaces at interactive rates; literally hundreds of times faster than ray tracing, yet with comparable accuracy.

The work performed on interactive methods for symbolic manipulation focused on the use of handwriting and gestures to enter and manipulate mathematical expressions. We have combined feature-based handwriting recognition and 2D equation parsing with several novel methods of interactive error correction to create a system that allows for rapid and natural freehand equation editing.

Finally, we have applied several useful techniques from computer vision to the problem of geometry capture for use in computer graphics. In particular, we have addressed the problem of extracting parameters for simple generative models from approximate range data captured through techniques such as "structured light."

2 Summary of Results

In the following sub-sections, we provide an overview of each of the four research categories addressed under this grant, as outlined above. Citations to the published papers are included in each section, and also summarized in section 3.

2.1 Distributed Monte Carlo Image Synthesis

Monte Carlo methods are the most versatile methods available for image synthesis, as they scale well to large or complex scenes, and can accommodate virtually any type of reflection phenomena. In fact, the ability to handle non-Lambertian reflection in the context of global illumination is perhaps the most compelling reason to employ Monte Carlo. Essentially the only drawback to these approaches is the excessively slow rate of convergence. Coupled with the high computational costs of simulating individual photon collisions and their resulting scattering distributions in complex environments, this slow convergence translates into massive computation.

Figure 1: *Two moderately complex models of interior scenes rendered using multi-processor Monte Carlo. Near optimal load balancing was performed using diffusion.*

In light of this, we have explored ways in which to efficiently perform large-scale Monte Carlo path tracing simulations (essentially photon transport simulations) for global illumination on massively parallel computer systems. The techniques investigated under this grant have focused on diffusion algorithms for load balancing, assuming that each processor has a complete copy of the geometric data. The diffusion scheme for propagating work to idle processors is applicable to arbitrary processor topologies and scales well to effectively arbitrary numbers of processors [6]. We have demonstrated the technique on scenes consisting of tens of thousands of polygons with diffuse, glossy, and mirror reflection [7]. Several of the test scenes that we employed are shown in Figure 1. These models were created and distributed as part of the "Radiance" rendering system of Ward [15].

Figure 2: *Ray traced scene with multiple reflections (left), and z-buffered scene computed using perturbation of virtual object (right). As the reflected object moves, the z-buffered image can be updated in 0.7 seconds, whereas the ray traced image requires 67 seconds to re-compute.*

A number of standard methods for improving the statistical efficiency of the estimators were incorporated into the rendering algorithm, including sample stratification and a simple form of importance sampling. These were added to the distributed algorithm very easily, and had essentially no impact on the effectiveness of the load balancing scheme, which distributed work in the form of rays to be traced, despite the fact they tended to dramatically change the distribution of the photon paths followed.

On architectures consisting of up to 256 processors, we have attained efficiencies as high as 99%. While Monte Carlo algorithms are inherently highly parallel, we have addressed many subtle difficulties encountered in image synthesis that tend to reduce efficiency, particularly in non-Lambertian scenes where there can be ray paths of dramatically different depths in different portions of the scene. We have shown that this characteristic can lead to relatively poor performance if work is simply distributed among processors randomly [8], which has been a popular method for distribuing photon simulations over many processors. We have shown that in many instances a diffusion algorithm will significantly out-perform random assignment.

Another characteristic of our approach is that very little of the prior history of a ray need be passed from one processor to the next; in particular, the forwarding processor is irrelevant. This allows non-recursive path tracing, in which the contribution of each scattering event is made without passing information back along the optical path (as is typical of recursion-based ray tracing). In a distributed setting, this is a great advantage as it dramatically reduces communication among processors, and allows for greater flexibility in diffusing the work load.

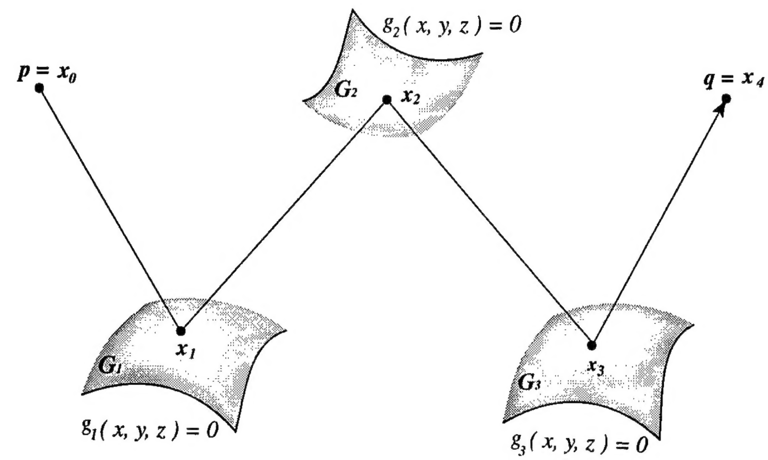

Figure 3: *An optical path resulting from specular reflection from three curves surfaces. The intermediate reflection points, x_1, x_2, and x_3, can be viewed as functions of the end points p and q.*

2.2 Perturbation Methods for Specular Reflection

Specular (i.e. mirror-like) reflection is an important special case of non-Lambertian reflection; in computer graphics, specular reflection is generally handled either by ray tracing or environment mapping, depending on whether exact or merely plausible reflections are required. Another aspect of the work performed under this grant was an exploration of new techniques for simulating specular reflection; we have demonstrated a new algorithm for computing nearly-perfect reflections in arbitrary curved surfaces at interactive rates [3]. A side-by-side comparison of an image generated using our technique and an image generated using standard ray tracing is shown in Figure 2.

Our technique is based on perturbation theory applied to optical paths [4]. In particular, the technique applies to paths followed by rays of light that reflect from one or more specular surfaces, as shown in Figure 3. Such a path is know to satisfy Fermat's principle of extremal distance; that is, the light follows a path that is either maximal or minimal in length. This property allows us to express the problem of finding such paths as one of constrained optimization. This, in turn, allows us to characterize such paths by means of Lagrange multipliers, which is precisely the approach taken by Mitchell and Hanrahan for computing illumination reflected from curved surfaces [9]. The central tool that we developed takes this a step further, and computes the first and second-order derivatives of the reflection path by means of the implicit function theorem. The result is a technique whereby closely related reflection paths can be computed via a Taylor series expansion about a know reflection path.

Figure 4 contrasts our approach with other techniques. In standard ray tracing, one

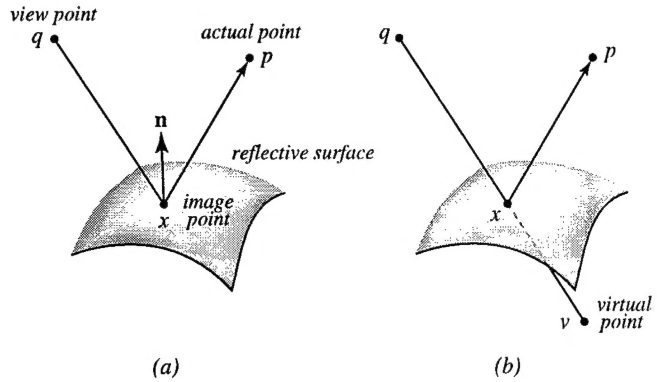

Figure 4: *Three different reflection problems, all with point q fixed: (a) find p given x (standard ray tracing), (b) find v given p, and (c) find the change in v given a change in p.*

seeks to solve the reflection problem shown in Figure 4(a): that is, given a ray and a point of reflection, we simply find the next point hit by the ray. In contrast, Mitchell and Hanrahan addressed the problem shown in Figure 4(b): that is, they computed the reflection points of given points in the environment, which is a significantly harder problem. Our approach relies upon a problem that is distinct from both of these: given an existing reflection path, we attempt to characterize how the reflection point(s) change as a result of moving the end point(s), as shown in Figure 4(c).

This new type of reflection problem is the key to generating high-quality reflections very rapidly. The central idea is depicted in Figure 5. Here, a virtual object is generated for each reflection of each real object. Our use of virtual objects is very similar to the approach taken by Ofek and Rappoport [10]. This virtual object is computed in such a way that its distance from the eye is the same as the original object's optical distance from the eye. This allows all virtual objects to be rendered correctly via z-buffering; that is, object occlusion is correctly modeled in all reflections. In this approach, reflectivity of surfaces is simulated by transparency. Consequently, z-buffering and alpha-blending become tools for rendering reflections.

The key to this algorithm is in computing the point(s) at which any given point in the environment is reflected in a curved mirror; with this information, the virtual object can be easily built. Our perturbation formula provides a very fast method for solving exactly this problem. As shown in Figure 5, the reflection paths that pass through the vertices of the real object are computed by perturbing one or more nearby paths. The nearby paths are generated via traditional ray tracing as a pre-processing step. All that is required is a very sparse sampling of reflection paths, so the pre-processing is extremely fast compared to a full ray tracing solution.

We have demonstrated this approach with an interactive program in which diffuse poly-

gons of a simple environment can be interactively moved. The reflections are then updated in real time by continually re-computing the virtual objects and re-rendering the resulting objects via a traditional graphics pipeline, which exploits both z-buffering and alpha blending. We have measured performance increases of several hundred over ray tracing, while sacrificing very little in accuracy.

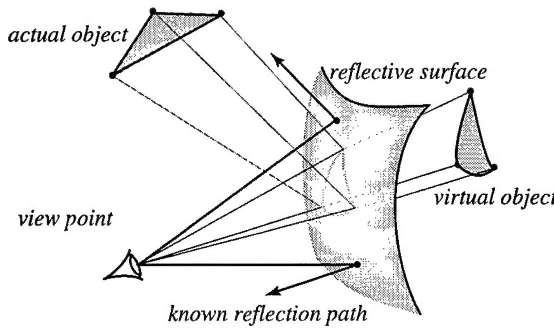

Figure 5: *The reflection of an actual object is approximated by forming a virtual object that maintains the same optical distance from the eye. The vertices of the virtual object are computed by perturbing actual reflection paths that pass near the desired ones.*

2.3 Interactive Symbolic Manipulation

One of the secondary objectives of this project was to address the problem of improving human-computer interaction in the context of symbolic mathematics. Toward that end, we have developed a prototype system for formula entry and editing that is purely handwriting and gesture driven [13]. Figure 6 shows a screen shot of our system, which offers a distinct and more natural alternative to the text-based and template-based approaches that are currently used for formula entry. In our system, the user writes as he/she would upon a white-board. The system uses character recognition and a 2D mathematical formula parser to extract the true syntax of the written formula. This extracted syntax can then be used as input to other packages, such as those for mathematical typesetting (as shown) or computer algebra (still in development).

This project is a component of a much longer-term project which is seeking to develop new and more natural environments for symbolic and numerical mathematics. Another investigation along these lines that was carried out under this grant targeted the use of diagrams for both discovery and explanation of mathematical ideas [1]. A hypothetical scenario is shown in Figure 7. In this scenario the computer generates illustrations that suggest various steps in a topology proof. These illustrations are intended to call out special cases and to suggest appropriate steps to follow in completing the proof. Other types of

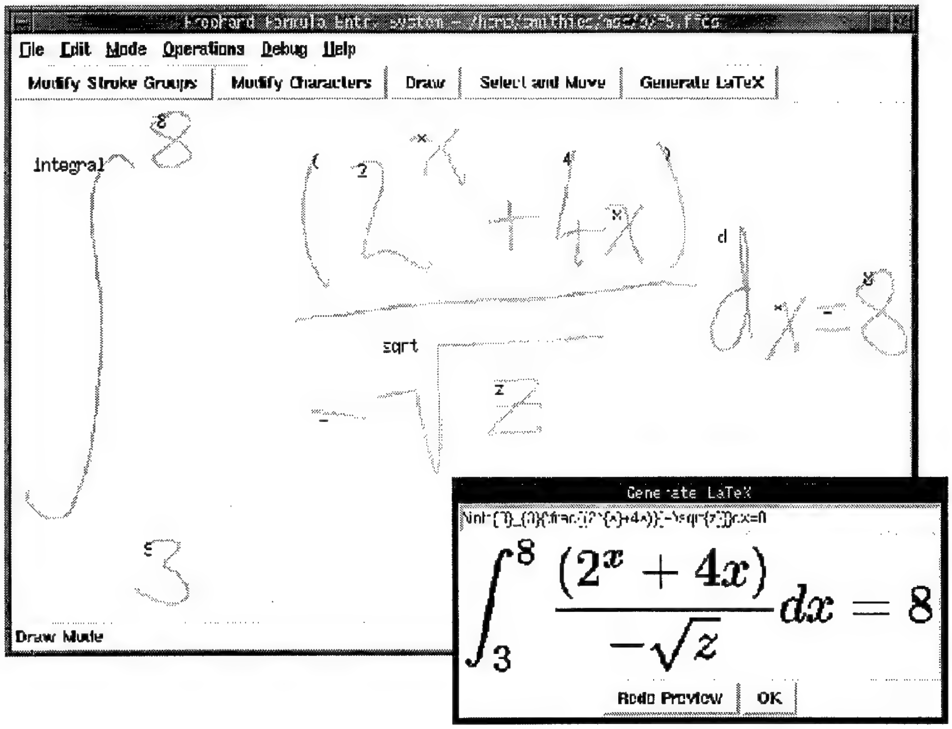

Figure 6: *A screen shot of the pen-based equation editor. As a user writes a formula, the strokes are grouped into characters, recognized, and passed to a two-dimensional parser. Once the formula is parsed, the output is typeset and shown in a separate window.*

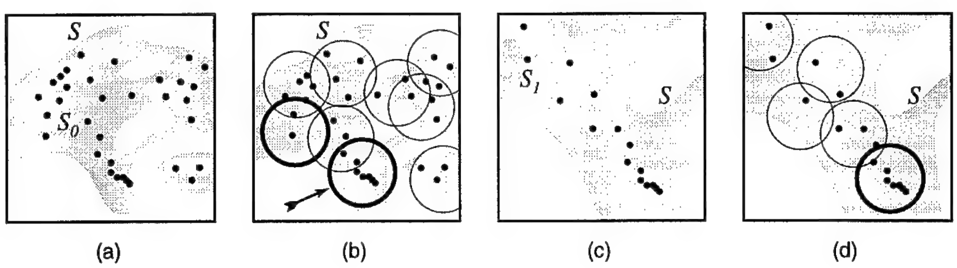

Figure 7: *A sequence of diagrams illustrating a new approach to diagrammatic interaction. Here the (hypothetical) computer-generated sequence of illustrations depicts the connection between two fundamental topological properties: compactness and total boundedness.*

diagrams considered in this work are are those with complete semantics, such as Venn diagrams and state diagrams for automata.

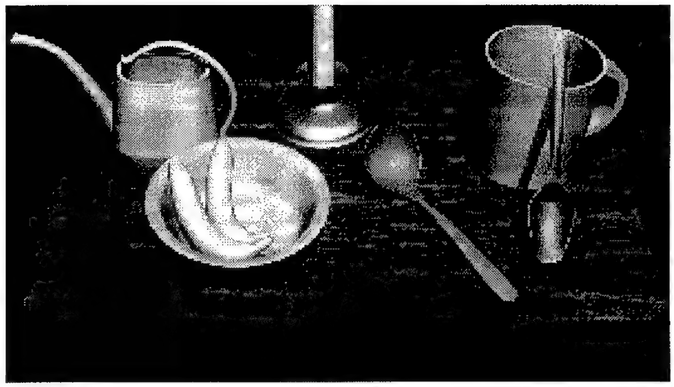

Figure 8: *A scene consisting of simple generative models that were acquired from noisy and incomplete range images. All of these shapes were modeled using only two different hierarchies of shape transformations.*

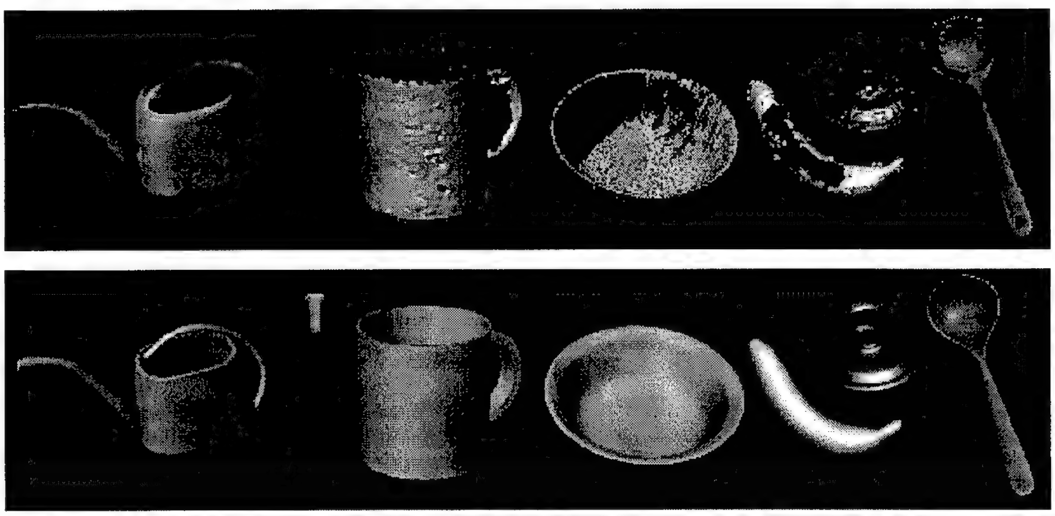

Figure 9: *The original range data used to determine the model parameters (top row), and the resulting generative models (bottom row). Note that the original range data is extremely noisy and incomplete, which would be unacceptable for approaches that extract meshes directly from the data.*

2.4 Recovering Geometry from Range Data

Finally, we have developed a technique for creating high-level parametric models from low-level range data. The technique can employ essentially any method for generating 3D point-clouds or range data from a real scene; such methods include mechanical probes,

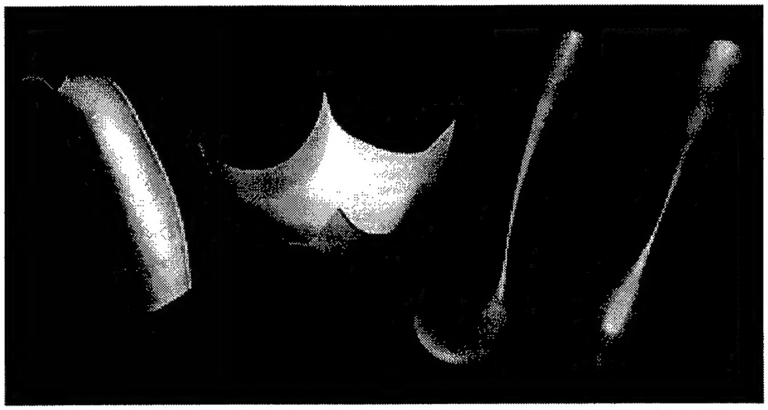

Figure 10: *Recovered models resulting from incorrect hierarchies. Here, each range image was fit to a model lacking the necessary degrees of freedom. Nonetheless, the approach was able to capture the most dominant features of the data, demonstrating the robustness of the approach.*

laser scanners, and structured light, all of which have been demonstrated to be effective alternatives in this context. Previous techniques for acquiring geometric descriptions from such point clouds have focused on creating meshes directly from the range data. However, these approaches involve large amounts of data which is difficult to manipulate, and are extremely sensitive to inaccurate or missing data.

In contrast, our approach is model-based. In particular, it is driven by a hierarchy of modeling operations, such as bending and twisting, that defines a parametric class of objects. Such a representation is known as a generative model [14]. Our technique searches for the best combination of modeling operations from within a given hierarchy, then optimizes the parameters associated with each operation to achieve the best fit possible to the range data. As a result, the model is completely determined by the composition of modeling operations that is selected coupled with the parameters. This representation is extremely compact; in fact, it is generally orders of magnitude smaller than an explicit mesh. More importantly, however, this representation fills in missing data and virtually eliminates noise. Figure 9 shows a comparison of raw data (top row) with the recovered models (bottom row).

The most significant limitation of our approach is that the model hierarchy must be pre-defined; the algorithm will only select models from within this hierarchy, which may in fact be inadequate to the task. Consequently, the approach is only viable for geometries for which there already exist reasonable generative models.

Figure 10 shows the result of applying inappropriate model hierarchies to acquired range data. In all cases the hierarchy fails to exhibit the necessary degrees of freedom to model the data. The left images are of the banana and bowl recovered using the hierarchy which is specific to a spoon, while the rotating generalized cylinder is used in the last two examples

to recover the ladle and spoon. Despite the mismatches, we see the algorithm nonetheless does the best it can, capturing the most dominant aspects of the shapes. This experiment demonstrates that the technique is quite robust; it will find a reasonable approximation, even within a very poorly chosen hierarchy.

3 Publications

The following papers have been published as a result of work conducted during the term of this grant. Complete citations can be found in the bibliography.

1. "Scalable photorealistic rendering of complex scenes," *First Eurographics Workshop on Parallel Graphics and Visualization*, 1996 [6]

2. "Scalable Monte Carlo image synthesis," *Parallel Computing*, 1997 [7]

3. "A competitive analysis of load balancing strategies for parallel ray tracing," *Journal of Supercomputing*, 1998 [8]

4. "Creating generative models from range images," *SIGGRAPH*, 1999 [12]

5. "A handwriting-based equation editor," *Graphics Interface*, 1999 [13]

6. "Computer aided serendipity: The role of autonomous assistants in problem solving," *Graphics Interface*, 1999 [1]

7. "Perturbation methods for interactive specular reflections," *Transactions on Visualization and Computer Graphics*, 2000 [3]

8. "Theory and application of specular path perturbation," *Transactions on Graphics*, 2000 [4]

4 Personnel

Four graduate students have been partially supported though this project: Anil Hirani, Min Chan, Alan Heirich, and Ravi Ramamoorthi. One Master's thesis (by Ravi Ramamoorthi) and one Ph.D. dissertation (by Alan Heirich) were completed during the term of this grant, and were partially supported by it. In addition, one Master's thesis (by Min Chen), completed in 1999, was partially supported by this grant. The theses are listed below:

1. Ravi Ramamoorthi, "Creating Generative Models from Range Images," Master's thesis, 1998 [11].

2. Alan Heirich, "Analysis of Scalable Algorithms for Dynamic Load Balancing and Mapping with Application to Photo-Realistic Rendering," Ph.D. dissertation, 1998 [5].

3. Min Chen, "Perturbation methods for image synthesis," Master's thesis, 1999 [2].

Full citations, including the technical report designations, can be found in the bibliography.

References

[1] James Arvo. Computer aided serendipity: The role of autonomous assistants in problem solving. In *Proceedings of Graphics Interface '99*, pages 183–192, Kingston, Ontario, June 1999.

[2] Min Chen. Perturbation methods for image synthesis. Master's thesis, California Institute of Technology, Pasadena, California, May 1999.

[3] Min Chen and James Arvo. Perturbation methods for interactive specular reflections. *IEEE Transactions on Visualization and Computer Graphics*, To Appear, 2000.

[4] Min Chen and James Arvo. Theory and application of specular path perturbation. *ACM Transactions on Graphics*, To Appear, 2000.

[5] Alan Heirich. *Analysis of Scalable Algorithms for Dynamic Load Balancing and Mapping with Application to Photo-Realistic Rendering.* PhD thesis, California Institute of Technology, Pasadena, California, June 1998. CS-TR-98-10.

[6] Alan Heirich and James Arvo. Scalable photorealistic rendering of complex scenes. In *Proceedings of the First Eurographics Workshop on Parallel Graphics and Visualization*, pages 17–33, Bristol, England, September 1996.

[7] Alan Heirich and James Arvo. Scalable Monte Carlo image synthesis. *Parallel Computing*, 23(7):845–859, July 1997.

[8] Alan Heirich and James Arvo. A competitive analysis of load balancing strategies for parallel ray tracing. *Journal of Supercomputing*, 12(1/2):57–68, 1998.

[9] Don Mitchell and Pat Hanrahan. Illumination from curved reflectors. *Computer Graphics*, 26(2):283–291, July 1992.

[10] Eyal Ofek and Ari Rappoport. Interactive reflections on curved objects. In *Computer Graphics* Proceedings, Annual Conference Series, ACM SIGGRAPH, pages 333–342, July 1998.

[11] Ravi Ramamoorthi. Creating generative models from range images. Master's thesis, California Institute of Technology, Pasadena, California, 1997.

[12] Ravi Ramamoorthi and James Arvo. Creating generative models from range images. In *Computer Graphics* Proceedings, Annual Conference Series, ACM SIGGRAPH, pages 195–204, August 1999.

[13] Steve Smithies, Kevin Novins, and James Arvo. A handwriting-based equation editor. In *Proceedings of Graphics Interface '99*, pages 84–91, Kingston, Ontario, June 1999.

[14] John M. Snyder. *Generative Modeling: An Approach to High Level Shape Design for Computer Graphics and CAD.* PhD thesis, California Institute of Technology, May 1991.

[15] Gregory J. Ward. The RADIANCE lighting simulation and rendering system. In *Computer Graphics* Proceedings, Annual Conference Series, ACM SIGGRAPH, pages 459–472, July 1994.